The ISC Int'l Journal of Information Security

ISeCure

January 2017, Volume 9, Number 1 (pp. 3–16)

http://www.isecure-journal.org

An Efficient Certificateless Signcryption Scheme in the Standard Model☆

Parvin Rastegari [1,*], and Mehdi Berenjkoub [1]

[1] *Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran*

ARTICLE INFO.

Article history:
Received: 24 December 2016
Revised: 25 January 2017
Accepted: 30 January 2017
Published Online: 31 January 2017

Keywords:
Certificateless Signcryption Scheme, Malicious-but-passive Key Generation Center Attack, Public Key Replacement Attack, Random Oracle Model, Standard Model.

ABSTRACT

Certificateless public key cryptography (CL-PKC) is a useful method in order to solve the problems of traditional public key infrastructure (i.e., large amount of computation, storage and communication costs for managing certificates) and ID-based public key cryptography (i.e., key escrow problem), simultaneously. A signcryption scheme is an important primitive in cryptographic protocols which provides the goals of signing and encrypting, simultaneously. In 2010, Liu *et al.* presented the first certificateless signcryption (CLSC) scheme in the standard model, but their scheme is vulnerable against different attacks presented in the literature, till now. In this paper, we improve their scheme and propose a new CLSC scheme, which is semantically secure against adaptive chosen ciphertext attack under the $(\mathcal{S}_2, 5)$-BDHE-Set assumption and existentially unforgeable against adaptive chosen message attack under the 3-CDHE assumption in the standard model. Our scheme is more efficient than all other secure CLSC schemes in the standard model proposed up to now.

1 Introduction

In a traditional public key cryptography (PKC), a user chooses a public/private key pair (pk, sk). In a conventional public key infrastructure (PKI), a certificate authority (CA) issues a digital certificate in order to bind between the public key and the identity of a user. However, the management of the certificates requires a large amount of computation, storage and communication costs. To avoid this problem, Shammir proposed the notion of identity-based cryptography (ID-PKC) in 1984 [1]. In an ID-PKC, a trusted third party, called the private key generator (PKG), generates the private key of a user from his unique identifier information. However, an inherent problem of ID-PKC is key escrow, i.e., the PKG knows all users' private keys. To eliminate these problems, simultaneously, Al Riyami and Paterson introduced the concept of certificateless public key cryptography (CL-PKC) in 2003 [2]. In a CL-PKC, a public/secret key pair (pk, x) is produced by the user himself without requiring pk to be certified. Also, a partial private key psk is generated by a semi-trusted third party, called key generation center (KGC), from the unique identifier information of the user. The user must know both x and psk to calculate his full private key sk. In other words, CL-PKC can be convinced as an intermediate between traditional PKI and ID-PKC.

Confidentiality, authentication, integrity and/or non-repudiation are some well-known security requirements in cryptographic protocols. Confidentiality can be achieved by encryption, and other mentioned requirements can be achieved by signing messages. When

☆ This article is an extended version of an ISCISC'13 paper.

* Corresponding author.
Email addresses: parvin.rastegari@ec.iut.ac.ir (P. Rastegari), brnjkb@cc.iut.ac.ir (M. Berenjkoub)

ISSN: 2008-2045

all these requirements must be satisfied at the same time, there are more efficient solutions than encrypting and signing each message, separately. A signcryption scheme is an important cryptographic primitive that provides the goals of encrypting and signing messages, simultaneously. Digital signcryption was first introduced by Zheng in 1997 [3].

The notion of a certificateless signcryption scheme was first introduced by Barbosa and Farshim in 2008 [4]. Since then many papers have appeared on certificateless signcryption about discussing different security models, cryptanalyzing the existing schemes, presenting new concrete schemes and proposing schemes for particular applications [5–14].

In 2010, Liu *et al.* presented a certificateless signcryption scheme and claimed that their scheme is the first scheme which satisfies the security requirements of a CLSC scheme, i.e., unforgeability and confidentiality against malicious -but-passive KGC and key replacement attacks in the standard model [5]. However, authors in [12] presented malicious -but-passive KGC attacks against both the confidentiality and unforgeability of Liu *et al.*'s CLSC scheme. Also, there exist key replacement attacks against confidentiality [13, 14] and unforgeability [13] of Liu *et al.*'s scheme, in the literature. In 2010, a supplement to Liu *et al.*'s CLSC scheme was proposed in [6] which is robust against the proposed attack in [14], but it is still vulnerable against the proposed attacks in [12]. Recently, in 2014, 2015 and 2016 three CLSC schemes in the standard model have been proposed in [7], [8] and [9], respectively which all of them seem secure against the attacks in [12–14]. However, these schemes are not so efficient. (Although the scheme in [8] is more efficient than the scheme in [7] and the scheme in [9] is more efficient than the scheme in [8].) Note that efficiency is a very important property in a signcryption scheme, since the main goal of a signcryption scheme is providing the goals of encryption and signature, simultaneously more efficient than encrypting and signing, separately [3].

In this paper, we will propose an improved version of Liu *et al.*'s CLSC scheme which is not only secure against the attacks in [12–14], but also more efficient than the proposed schemes in [7–9]. The security of our proposed scheme is provable in the standard model (without random oracles).

The rest of this paper is organized as follows: In Section 2, the formal model and the adversarial models of a certificateless signcryption scheme will be introduced. In Section 3, we will present our improved scheme. Section 4, the security and the performance of the proposed scheme will be analyzed. Finally, we will conclude the paper in Section 5.

2 Certificateless Signcryption

2.1 Formal Model

A generic certificateless signcryption scheme, involves three entities: a key generation centre (KGC), a sender (S) and a receiver (R) and is defined by six algorithms as follows:

- **Setup.** It is a probabilistic polynomial time (PPT) algorithm which takes as input a security parameter k and outputs system parameters $params$ and a master secret key msk. After running this algorithm, the KGC publishes $params$ and keeps msk secret.
- **Partial-Private-Key-Generation.** It is a PPT algorithm which takes as input $params$, msk and an identity $ID \in \{0,1\}^*$ and outputs a partial private key psk_{ID}. The KGC runs this algorithm and sends psk_{ID} to the corresponding entity via a secure channel.
- **User-Key-Generation.** It is a PPT algorithm which takes as input $params$ and ID and outputs a randomly selected value x_{ID} and a corresponding public key pk_{ID}. After running this algorithm by the entity with identity ID, he keeps x_{ID} secret and publishes pk_{ID} without requiring to be certified.
- **Private-Key-Generation.** It is a PPT algorithm which takes as input $params$, psk_{ID} and x_{ID} and outputs the entity's full private key sk_{ID}. This algorithm is executed by the entity, himself.
- **Signcryption.** It is a PPT algorithm which takes as input $params$, a message M, the sender's private key sk_{ID_S}, the receiver's public key pk_{ID_R}, and outputs a signcryption σ on message M.
- **Unsigncryption.** It is a deterministic polynomial time algorithm which takes as input $params$, a signcryption σ, the receiver's private key sk_{ID_R}, the sender's public key pk_{ID_S}, and outputs M if the signature is valid and $\perp$ otherwise.

Note that the correctness must be satisfied, i.e., if $\sigma = Signcrypt(params, M, sk_{ID_S}, pk_{ID_R})$, then the output of $Unsigncrypt(params, \sigma, sk_{ID_R}, pk_{ID_S})$ must contain M and a guarantee that M is actually signcrypted by the sender S.

2.2 Adverserial Models and Oracle Accesses

In a certificateless public key cryptography two types of adversaries are considered [5]. A type I adversary $\mathcal{A}_I$ models an adversary who can replace the public key of an arbitrary entity, but does not access to the master secret key (key replacement attack). A type II adversary $\mathcal{A}_{II}$ models an adversary who possesses the master secret key but cannot replace any public keys. In [5], $\mathcal{A}_{II}$ is considered as a malicious-but-passive KGC, who can try to decrypt a ciphertext or forge a signature (malicious-but-passive KGC attack).

During an attack against a cryptographic scheme, the adversary can obtain some information from the environment. This is modelled by some oracles that the adversary can send some requests to them. In a certificateless signcryption scheme, the adversaries may access to the following oracles:

- $\mathcal{O}_{pk}$. Refers to the public key oracle, which takes as input an identity ID and outputs the corresponding public key pk_{ID}.
- $\mathcal{O}_{psk}$. Refers to the partial private key oracle, which takes as input an identity ID and outputs psk_{ID}.
- $\mathcal{O}_{Replace.pk}$. Refers to the replaced public key oracle, which takes as input an identity ID and a new valid public key pk'_{ID} and replaces pk_{ID} with pk'_{ID}.
- $\mathcal{O}_{sk}$. Refers to the private key oracle, which takes as input an identity ID and outputs the corresponding private key sk_{ID} for the identity whose public key has not been replaced.
- $\mathcal{O}_{Signcrypt}$. Refers to the signcryption oracle, which takes as input M, ID_S, ID_R and outputs a valid signcryption σ.
- $\mathcal{O}_{Unsigncrypt}$. Refers to the unsigncryption oracle, which takes as input σ, ID_S, ID_R and outputs the results of the Unsigncryption algorithm.

Note that a type I adversary, i.e., $\mathcal{A}_I$, has access to all of the above oracles and a type II adversary, i.e., $\mathcal{A}_{II}$, has access to all of the above oracles except $\mathcal{O}_{Replace.pk}$ and $\mathcal{O}_{psk}$ (Because $\mathcal{A}_{II}$ herself possesses the msk and can generate psk, so she does not need to has queries from $\mathcal{O}_{psk}$). In the rest of this paper, $\mathcal{O}_I = \{\mathcal{O}_{pk}, \mathcal{O}_{psk}, \mathcal{O}_{Replace.pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$ is the set of all oracles which can be accessed by $\mathcal{A}_I$ and $\mathcal{O}_{II} = \{\mathcal{O}_{pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$ is the set of all oracles which can be accessed by $\mathcal{A}_{II}$.

2.3 Security Requirements

Confidentiality and unforgeability are two main security requirements for a signcryption scheme. In this section, these security requirements will be described via some games between an adversary ($\mathcal{A}_I$ or $\mathcal{A}_{II}$) and a challenger $\mathcal{C}$. In the following games some notations will be used which we describe them here:

- $Outputs \longleftarrow X_{Al}(Inputs)$: The entity X runs the algorithm Al on $Inputs$ and generates $Outputs$.
- $Q = \{res_1, res_1, \ldots res_q\} \longleftarrow Queries(X, \mathcal{O})$: The entity X sends q (a polynomially bounded number) queries to the set of oracles $\mathcal{O}$ and obtains Q as response. Note that all queries can be made adaptively, i.e., each query may depend on the answers to the previous queries.
- $\gamma \longleftarrow X_R$: The entity X randomly selects a bit $\gamma \in_R \{0, 1\}$.

Confidentiality. This property is considered as indistinguishability of encryptions under the adaptive chosen ciphertext attack (IND-CCA) and is defined by Game I and Game II for type I and type II adversaries, respectively.

Figure 1. Game I

Initialization:

$(msk, params) \longleftarrow \mathcal{C}_{Setup}(k)$

($\mathcal{C}$ keeps msk secret and gives $params$ to $\mathcal{A}_I$.)

Phase 1 queries:

$Q_1 \longleftarrow Queries(\mathcal{A}_I, \mathcal{O}_I)$

Challenge:

$(\{ID_{S^*}, ID_{R^*}\}, \{M_0, M_1\}) \longleftarrow \mathcal{A}_I(Q_1)$

$\gamma \longleftarrow \mathcal{C}_R(\{ID_{S^*}, ID_{R^*}\}, \{M_0, M_1\})$

$\sigma^* \longleftarrow \mathcal{C}_{Signcrypt}(params, M_\gamma, sk_{ID_{S^*}}, pk_{ID_{R^*}})$

($\mathcal{C}$ sends σ^* to $\mathcal{A}_I$.)

Phase 2 queries:

$Q_2 \longleftarrow Queries(\mathcal{A}_I, \mathcal{O}_I)$

Response:

$\gamma^* \longleftarrow \mathcal{A}_I(Q_1, Q_2, \sigma^*)$

It is said that $\mathcal{A}_I$ wins Game I if $\gamma^* = \gamma$ and the following conditions hold:

(1) $\mathcal{A}_I$ cannot extract $sk_{ID_{R^*}}$ at any point.
(2) $\mathcal{A}_I$ cannot extract sk_{ID} for any identity if the corresponding public key has already been replaced.
(3) $\mathcal{A}_I$ cannot extract $psk_{ID_{R^*}}$ if $\mathcal{A}_I$ has replaced $pk_{ID_{R^*}}$ before the challenge step.
(4) In phase 2 queries, $\mathcal{A}_I$ cannot make an unsigncryption query on σ^* under ID_{S^*} and ID_{R^*}, unless $pk_{ID_{S^*}}$ or $pk_{ID_{R^*}}$ used to signcrypt M_γ has been replaced after the challenge was issued.

It is said that $\mathcal{A}_{II}$ wins Game II if $\gamma^* = \gamma$ and the following conditions hold:

(1) $\mathcal{A}_{II}$ cannot extract $sk_{ID_{R^*}}$ at any point.
(2) In phase 2 queries, $\mathcal{A}_{II}$ cannot make an unsigncryption query on σ^* under ID_{S^*} and ID_{R^*}.

Definition 1. A certificateless signcryption scheme is $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$-semantically secure under adaptive chosen ciphertext attack if no adversaries ($\mathcal{A}_I$ and $\mathcal{A}_{II}$) with at most running time t, making at most q_{pk} public key queries from $\mathcal{O}_{pk}$, q_{psk} partial private key queries from $\mathcal{O}_{psk}$ ($q_{psk} = 0$ for $\mathcal{A}_{II}$), q_{Rpk} public key replacement queries from $\mathcal{O}_{Replace.pk}$ ($q_{Rpk} = 0$ for $\mathcal{A}_{II}$), q_{sk} private key queries from $\mathcal{O}_{sk}$, q_S signcryption queries from $\mathcal{O}_{Signcrypt}$ and q_U un-

Figure 2. Game II

Initialization:

$(msk, params) \longleftarrow \mathcal{A}_{II_{Setup}}(k)$

($\mathcal{A}_{II}$ gives msk and $params$ to $\mathcal{C}$.)

Phase 1 queries:

$Q_1 \longleftarrow Queries(\mathcal{A}_{II}, \mathcal{O}_{II})$

Challenge:

$(\{ID_{S^*}, ID_{R^*}\}, \{M_0, M_1\}) \longleftarrow \mathcal{A}_{II}(Q_1)$

$\gamma \longleftarrow \mathcal{C}_R(\{ID_{S^*}, ID_{R^*}\}, \{M_0, M_1\})$

$\sigma^* \longleftarrow \mathcal{C}_{Signcrypt}(params, M_\gamma, sk_{ID_{S^*}}, pk_{ID_{R^*}})$

($\mathcal{C}$ sends σ^* to $\mathcal{A}_{II}$.)

Phase 2 queries:

$Q_2 \longleftarrow Queries(\mathcal{A}_{II}, \mathcal{O}_{II})$

Response:

$\gamma^* \longleftarrow \mathcal{A}_{II}(Q_1, Q_2, \sigma^*)$

Figure 3. Game III

Initialization:

$(msk, params) \longleftarrow \mathcal{C}_{Setup}(k)$

($\mathcal{C}$ keeps msk secret and gives $params$ to $\mathcal{A}_I$)

Queries:

$Q \longleftarrow Queries(\mathcal{A}_I, \mathcal{O}_I)$

Output:

$(\sigma^*, ID_{S^*}, ID_{R^*}) \longleftarrow \mathcal{A}_I(Q)$

(σ^* is not produced by the signcryption oracle)

signcryption queries from $\mathcal{O}_{Unsigncrypt}$, wins Game I and Game II with probability at least $\frac{1}{2} + \varepsilon$.

Unforgeability. This property is considered as existential unforgeability against chosen message attack (EUF-CMA) and is defined by Game III and Game IV for type I and type II adversaries, respectively.

It is said that $\mathcal{A}_I$ wins Game III if the result of checking the signature in

$$Unsigncrypt(params, \sigma^*, sk_{ID_{R^*}}, pk_{ID_{S^*}})$$

is valid and the following conditions hold:

(1) $\mathcal{A}_I$ cannot extract $sk_{ID_{R^*}}$ at any point.
(2) $\mathcal{A}_I$ cannot extract sk_{ID} for any identity if the corresponding public key has already been replaced.
(3) $\mathcal{A}_I$ cannot extract $psk_{ID_{R^*}}$.

It is said that $\mathcal{A}_{II}$ wins Game IV if the result of checking the signature in

$$Unsigncrypt(params, \sigma^*, sk_{ID_{R^*}}, pk_{ID_{S^*}})$$

is valid and $\mathcal{A}_{II}$ cannot extract $sk_{ID_{R^*}}$ at any point.

Figure 4. Game IV

Initialization:

$(msk, params) \longleftarrow \mathcal{A}_{II_{Setup}}(k)$

($\mathcal{A}_{II}$ gives msk and $params$ to $\mathcal{C}$)

Queries:

$Q \longleftarrow Queries(\mathcal{A}_{II}, \mathcal{O}_{II})$

Output:

$(\sigma^*, ID_{S^*}, ID_{R^*}) \longleftarrow \mathcal{A}_{II}(Q)$

(σ^* is not produced by the signcryption oracle)

Definition 2. A certificateless signcryption scheme is $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$-existentially unforgeable under adaptive chosen message attack if no adversaries ($\mathcal{A}_I$ and $\mathcal{A}_{II}$) running time at most t, making at most q_{pk} public key queries from $\mathcal{O}_{pk}$, q_{psk} partial private key queries from $\mathcal{O}_{psk}$ ($q_{psk} = 0$ for $\mathcal{A}_{II}$), q_{Rpk} public key replacement queries from $\mathcal{O}_{Replacepk}$ ($q_{Rpk} = 0$ for $\mathcal{A}_{II}$), q_{sk} private key queries from $\mathcal{O}_{sk}$, q_S signcryption queries from $\mathcal{O}_{Signcrypt}$ and q_U unsigncryption queries from $\mathcal{O}_{Unsigncrypt}$, wins Game III and Game IV with probability at least ε.

3 Our Proposed Scheme

3.1 Biliniear Pairings

Let G_1 and G_2 be two multiplicative cyclic groups of prime order q and let g be a generator of G_1. There exists an admissible bilinear pairing $e : G_1 \times G_1 \longrightarrow G_2$ if and only if the following properties are satisfied.

(1) Bilinearity: $e(g^a, g^b) = e(g, g)^{ab}$, for all $a, b \in Z_q^*$.
(2) Non-degeneracy: i.e., $e(g, g) \neq 1_{G_2}$.
(3) Computability: There exists an efficient algorithm for computing $e(g, g)$.

3.2 Related Complexity Assumptions

In this subsection, some required assumptions in complexity theory are described.

Definition 3. $K+1$-Computational Diffie-Hellman Exponent ($K+1$-CDHE) problem [15] is that on inputs $g, g^a, \ldots, g^{a^K} \in G_1$, for unknown $a \in Z_q^*$, calculate $g^{a^{K+1}}$. It is said that (ε, t)-$K+1$-CDHE assumption holds in G_1, if no t-time algorithm can solve the $K+1$-CDHE problem in G_1, with probability at least ε.

Remark 1. The unforgeability of our proposed

scheme against $\mathcal{A}_I$ and $\mathcal{A}_{II}$ are based on 2-CDHE and 3-CDHE assumptions, respectively.

Definition 4. Consider a set of integers $\mathcal{S} \subset Z$, and define $\mathcal{S} +_q \mathcal{S} \triangleq \{i + j \bmod \lambda(q) : i, j \in \mathcal{S}\}$, where $\lambda(q)$ is the order of elements modulo q. Also, consider another target integer $m \notin S +_q \mathcal{S}$. The $(\mathcal{S}, m)$-Bilinear Diffie-Hellman Exponent-Set ($(\mathcal{S}, m)$-BDHE-Set) problem [16] is that on inputs $\{g^{a^i} \in G_1 : i \in \mathcal{S}\}$, for unknown $a \in Z_q^*$, and $X \in G_2$, decide whether $X = e(g, g)^{a^m}$. It is said that (ε, t)-$(\mathcal{S}, m)$-BDHE-Set assumption holds in (G_1, G_2), if no t-time algorithm can solve the $(\mathcal{S}, m)$-BDHE-Set problem in (G_1, G_2), with probability at least $\frac{1}{2} + \varepsilon$.

Remark 2. Assigning $\mathcal{S} = \mathcal{S}_K = \{0, 1, 2, \, dots, K\}$, and $m = 2K + 1$, the $(\mathcal{S}, m)$-BDHE-Set problem is that on inputs $g, g^a, g^{a^2}, \ldots, g^{a^K} \in G_1$, for unknown $a \in Z_q^*$, and $X \in G_2$ decide whether $X = e(g, g)^{a^{2K+1}}$. The confidentiality of our proposed scheme against $\mathcal{A}_I$ and $\mathcal{A}_{II}$ are based on $(\mathcal{S}_1, 3)$-BDHE-Set and $(\mathcal{S}_2, 5)$-BDHE-Set assumptions, respectively.

3.3 Proposed Scheme

In this subsection we present our certificateless signcryption scheme which is an improvement of Liu *et al.*'s scheme [5]. We use the certificateless signature scheme in [17] as the base of our signcryption scheme. Table 1 shows some symbols and notations which are used in our scheme.

For generality, identities can be considered of arbitrary lengths and a hash function $H_u : \{0,1\}^* \longrightarrow \{0,1\}^{n_u}$ can be used to convert them to the specific length n_u. The algorithms of our scheme are as follows:

- **Setup:** Given k, the KGC selects two multiplicative cyclic groups G_1 and G_2 of a large prime order q, a random generator g of G_1 and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$. It also chooses a random $\alpha \in Z_q^*$ and sets $g_1 = g^\alpha$ and $T = e(g_1, g_1)$. Furthermore, it selects two random values $u', v' \in G_1$ and two random vectors $U = (u_i) \in G_1^{n_u}$ and $V = (v_j) \in G_1^{n_m}$. The KGC also selects two collision resistant hash functions $H_1 : \{0,1\}^* \longrightarrow \{0,1\}^{n_m}$ and $H_2 : \{0,1\}^* \longrightarrow Z_q^*$. The public system parameters are $params = \{G_1, G_2, q, e, g, g_1, T, u', v', U, V, H_1, H_2\}$ and the master secret key is $msk = g^{\alpha^2}$.
- **Partial-Private-Key-Generation:** The KGC randomly selects $r \in Z_q^*$ and computes psk_{ID}, as follows:

$$psk_{ID} = (psk_{ID,1}, psk_{ID,2}) = (g^{\alpha^2}(u' \prod_{i \in \mathcal{U}_{ID}} u_i)^r, g^r).$$

- **User-Key-Generation:** The user with identity ID, selects a random secret value $x_{ID} \in Z_q^*$ as his secret key and computes the corresponding public key as

$$pk_{ID} = (pk_{ID,1}, pk_{ID,2}) = (g_1^{x_{ID}}, g_1^{\frac{1}{x_{ID}}}).$$

- **Private-Key-Generation:** The user with identity ID, selects a random value $r' \in Z_q^*$ and computes his full private key as

$$sk_{ID} = (sk_{ID,1}, sk_{ID,2}) = (psk_{ID,1}^{x_{ID}^2}(u' \prod_{i \in \mathcal{U}_{ID}} u_i)^{r'}, psk_{ID,2}^{x_{ID}^2} g^{r'}).$$

- **Signcryption:** Suppose that the sender with identity ID_S, wants to send a message $M \in G_2$ to the receiver with identity ID_R. The sender selects random values $r_1, r_2 \in Z_q^*$ and runs the following steps:
  (1) Checks whether $e(pk_{ID_R,1}, pk_{ID_R,2}) = T$ holds or not. If the equality does not hold, aborts and outputs $\perp$.
  (2) Computes $\sigma_1 = M.e(pk_{ID_R,1}^{r_1}, pk_{ID_R,1}) = M.e(g_1, g_1)^{x_{ID_R}^2 r_1}$.
  (3) Computes $\sigma_2 = g^{r_1}$.
  (4) Computes $\sigma_3 = (u' \prod_{i \in \mathcal{U}_{ID_R}} u_i)^{r_1}$.
  (5) Computes $\sigma_4 = sk_{ID_S,2}.g^{r_2}$.
  (6) Computes $m = H_1(\sigma_1, \sigma_2, \sigma_3, \sigma_4, ID_R, pk_{ID_R,1}) \in \{0,1\}^{n_m}$, and also computes $\mathcal{M} = \{j | m[j] = 1, j = 1, 2, \ldots, n_m\}$.
  (7) Computes $h = H_2(ID_S, m, pk_{ID_S,1}, \sigma_4, \sigma_2)$.

Table 1. Some notations in our scheme

Notation	Description
n_u	The length of identities
k	The security parameter
G_1 and G_2	Two multiplicative cyclic groups of a large prime order q
g	A random generator of G_1
$e : G_1 \times G_1 \longrightarrow G_2$	A bilinear pairing
$H_1 : \{0,1\}^* \longrightarrow \{0,1\}^{n_m}$ $H_2 : \{0,1\}^* \longrightarrow Z_q^*$	Two collision resistant hash functions
α	A random value in Z_q^*
u', v'	Two random values in G_1
$U = (u_i) \in G_1^{n_u}$	A random vector of length n_u that its elements u_is ($i = 1, 2, \ldots, n_u$) are chosen randomly from G_1
$V = (v_j) \in G_1^{n_m}$	A random vector of length n_m that its elements v_js ($j = 1, 2, \ldots, n_m$) are chosen randomly from G_1
msk	The master secret key
$ID[i]$	The i-th bit of an identity ID
$\mathcal{U}_{ID}$	$\mathcal{U}_{ID} = \{i \mid ID[i] = 1, i = 1, 2, \ldots, n_u\}$
psk_{ID}	The partial private key of a user with identity ID
$x_{ID} \in Z_q^*$	The secret key of a user with identity ID
pk_{ID}	The public key of a user with identity ID
sk_{ID}	The full private key of a user with identity ID
m	The H_1 value of a string which will be described in the signcription phase
$m[j]$	The j-th bit of $m \in \{0,1\}^{n_m}$
$\mathcal{M}$	$\mathcal{M} = \{j \mid m[j] = 1, j = 1, 2, \ldots, n_m\}$

(8) Computes $\sigma_5 = sk_{ID_S,1}.(u'\prod_{i\in\mathcal{U}_{ID_S}} u_i)^{r_2}$ $.(pk^h_{ID_S,1}(v'\prod_{j\in\mathcal{M}} v_j))^{r_1}$.
(9) Outputs $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$.

- **Unsigncryption:** The receiver checks the validity and decrypts $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$ as follows:
(1) Checks whether $e(pk_{ID_S,1}, pk_{ID_S,2}) = T$ holds or not. If the equality does not hold, aborts and outputs $\perp$.
(2) Computes $m = H_1(\sigma_1, \sigma_2, \sigma_3, \sigma_4, ID_R, pk_{ID_R,1}) \in \{0,1\}^{n_m}$, and also computes $\mathcal{M} = \{j | m[j] = 1, j = 1, 2, \ldots, n_m\}$.
(3) Computes $h = H_2(ID_S, m, pk_{ID_S,1}, \sigma_4, \sigma_2)$.
(4) Verifies the equality

$$\begin{aligned} e(\sigma_5, g) =& e(pk_{ID_S,1}, pk_{ID_S,1}) \\ & .e(u' \prod_{i\in\mathcal{U}_{ID_S}} u_i, \sigma_4) \\ & .e(pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j), \sigma_2). \end{aligned}$$

If the equally holds, computes and outputs

$$M = \sigma_1 . \frac{e(\sigma_3, sk_{ID_R,2})}{e(\sigma_2, sk_{ID_R,1})},$$

else outputs $\perp$.

4 Analysis of the Proposed Scheme

4.1 Correctness

The correctness can be simply verified as:

$$\begin{aligned} e(\sigma_5, g) =& e(sk_{ID_S,1}, g) \\ & .e((u' \prod_{i\in\mathcal{U}_{ID_S}} u_i)^{r_2}, g) \\ & .e((pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j))^{r_1}, g) \\ =& e(g^{\alpha^2 x^2_{ID_S}} (u' \prod_{i\in\mathcal{U}_{ID_S}} u_i)^{r x^2_{ID_S} + r'}, g) \\ & .e((u' \prod_{i\in\mathcal{U}_{ID_S}} u_i)^{r_2}, g) \\ & .e((pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j))^{r_1}, g) \\ =& e(g^{\alpha^2 x^2_{ID_S}}, g) \\ & .e((u' \prod_{i\in\mathcal{U}_{ID_S}} u_i)^{r x^2_{ID_S} + r' + r_2}, g) \\ & .e((pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j))^{r_1}, g) \\ =& e(g_1^{x_{ID_S}}, g_1^{x_{ID_S}}) \\ & .e(u' \prod_{i\in\mathcal{U}_{ID_S}} u_i, sk_{ID_S,2} g^{r_2}) \\ & .e(pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j), g^{r_1}) \\ =& e(pk_{ID_S,1}, pk_{ID_S,1}) \\ & .e(u' \prod_{i\in\mathcal{U}_{ID_S}} u_i, \sigma_4) \\ & .e(pk^h_{ID_S,1}(v' \prod_{j\in\mathcal{M}} v_j), \sigma_2), \end{aligned}$$

and

$$\begin{aligned} & \frac{\sigma_1 . e(\sigma_3, sk_{ID_R,2})}{e(\sigma_2, sk_{ID_R,1})} \\ =& \frac{M . e(g_1, g_1)^{x^2_{ID_R} r_1} . e((u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r_1}, g^{r x^2_{ID_R} + r'})}{e(g^{r_1}, g^{\alpha^2 x^2_{ID_R}} (u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r x^2_{ID_R} + r'})} \\ =& \frac{M . e(g^\alpha, g^\alpha)^{x^2_{ID_R} r_1} . e((u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r_1}, g^{r x^2_{ID_R} + r'})}{e(g^{r_1}, g^{\alpha^2 x^2_{ID_R}}) . e(g^{r_1}, (u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r x^2_{ID_R} + r'})} \\ =& \frac{M . e(g, g)^{\alpha^2 x^2_{ID_R} r_1} e((u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r_1}, g^{r x^2_{ID_R} + r'})}{e(g, g)^{\alpha^2 x^2_{ID_R} r_1} . e(g^{r x^2_{ID_R} + r'}, (u' \prod_{i\in\mathcal{U}_{ID_R}} u_i)^{r_1})} \\ =& M \end{aligned}$$

4.2 Security Analysis

The security of the proposed scheme is provable in the standard model. Define:

$$\begin{aligned} \varepsilon_I &\triangleq \frac{1}{8q_U(q_{psk} + q_{sk} + q_S + q_U + 1)(n_m + 1)(n_u + 1)}, \\ t_I &\triangleq order(((q_{psk} + q_{sk} + q_S + q_U)n_u + (q_S + q_U)n_m)T_M \\ &\quad + (q_{pk} + q_{psk} + q_{sk} + q_S + q_U)T_E + (q_S + q_U)T_P), \\ \varepsilon_{II} &\triangleq \frac{1}{8q_U(q_{sk} + q_S + q_U + 1)(n_m + 1)(n_u + 1)}, \\ t_{II} &\triangleq order(((q_{sk} + q_S + q_U)n_u + (q_S + q_U)n_m)T_M \\ &\quad + (q_{pk} + q_{sk} + q_S + q_U)T_E + (q_S + q_U)T_P), \end{aligned}$$

where T_M, T_E and T_P are the time for multiplication and exponentiation in G_1 and a pairing computation, respectively.

Lemma 1. The proposed scheme is $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$-semantically secure against $\mathcal{A}_I$, if the (ε', t')-$(\mathcal{S}_1, 3)$-BDHE-Set assumption holds in (G_1, G_2), where $\varepsilon' \geq \varepsilon\varepsilon_I$ and $t' \leq t + t_I$.

Proof. Suppose that there exists a $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$-type I adversary $\mathcal{A}_I$, who can break the indistinguishability of encryptions against adaptive chosen ciphertext attack (IND-CCA) in the proposed scheme according to Game I. By this assumption, we can construct a simulator $\mathcal{B}$ that can use $\mathcal{A}_I$ as a sub-routine and solve the $(\mathcal{S}_1, 3)$-BDHE-Set problem with a probability at least ε' and in time at most t', which contradicts the (ε', t')-$(\mathcal{S}_1, 3)$-BDHE-Set assumption in (G_1, G_2).

Consider two multiplicative cyclic groups G_1 and G_2 of a large prime order q, a random generator g of G_1 and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$. Suppose

that $\mathcal{B}$ is given a random $(\mathcal{S}_1, 3)$-BDHE-Set challenge ($g \in G_1, A = g^a \in G_1, X \in G_2$) and outputs a guess $\beta' = 1$, if he decides that $X = e(g,g)^{a^3}$ and $\beta' = 0$, otherwise. In order to use $\mathcal{A}_I$ as a subroutine, $\mathcal{B}$ must simulate the challenger $\mathcal{C}$ and answer all $\mathcal{A}_I$'s queries in Game I. In order to respond these queries consistently, $\mathcal{B}$ creates a database $\mathcal{DB} = \{(ID, psk_{ID}, x_{ID}, pk_{ID}, sk_{ID}, sta = 0)\}$ which is initially empty. Then $\mathcal{B}$ plays Game I with $\mathcal{A}_I$ and simulates $\mathcal{C}$ and all oracles which $\mathcal{A}_I$ has access to them, i.e., $\mathcal{O}_I = \{\mathcal{O}_{pk}, \mathcal{O}_{psk}, \mathcal{O}_{Replace.pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$, as follows:

Initialization: Let $l_u = 2(q_{psk}+q_{sk}+q_S+q_U+1)$ and $l_m = 2q_U$. Assume that $l_u(n_u+1) < q$ and $l_m(n_m+1) < q$. $\mathcal{B}$ randomly selects the following elements:

- $k_u \in_R \{0,1,\ldots,n_u\}$ and $k_m \in_R \{0,1,\ldots,n_m\}$. (By the assumptions of $l_u(n_u+1) < q$ and $l_m(n_m+1) < q$, we have $0 \le k_u l_u < q$ and $0 \le k_m l_m < q$.)
- $x', x_1, \ldots, x_{n_u} \in_R Z_{l_u}$ and $y', y_1, \ldots, y_{n_u} \in_R Z_q$.
- $z', z_1, \ldots, z_{n_m} \in_R Z_{l_m}$ and $w', w_1, \ldots, w_{n_m} \in_R Z_q$.

These values are kept internal to $\mathcal{B}$. Then $\mathcal{B}$ assigns a set of public parameters as follows:

$$
\begin{aligned}
&g_1 = A = g^a,\\
&u' = g_1^{-k_u l_u + x'} g^{y'},\\
&u_i = g_1^{x_i} g^{y_i} \quad (i = 1,2,\ldots,n_u),\\
&v' = g_1^{-k_m l_m + z'} g^{w'},\\
&v_j = g_1^{z_j} g^{w_j} \quad (j = 1,2,\ldots,n_m).
\end{aligned}
$$

$\mathcal{B}$ also computes $T = e(g_1, g_1)$ and selects two collision resistant hash functions $H_1 : \{0,1\}^* \longrightarrow \{0,1\}^{n_m}$ and $H_2 : \{0,1\}^* \longrightarrow Z_q^*$ and gives $params = \{G_1, G_2, q, e, g, g_1, T, u', v', U, V, H_1, H_2\}$ to $\mathcal{A}_I$. From the perspective of $\mathcal{A}_I$, all distributions are identical to those in the real world.

In order to follow the proof more easily, define four following functions:

$$
\begin{aligned}
J_u(ID) &= x' + \sum_{i \in \mathcal{U}_{ID}} x_i - k_u l_u,\\
K_u(ID) &= y' + \sum_{i \in \mathcal{U}_{ID}} y_i,\\
J_m(m) &= z' + \sum_{j \in \mathcal{M}} z_j - k_m l_m,\\
K_m(m) &= w' + \sum_{j \in M} w_j,
\end{aligned}
$$

where $\mathcal{U}_{ID}$ and $\mathcal{M}$ are defined similar to those in the proposed scheme. By These assignments, the following equations hold:

$$
\begin{aligned}
u' \prod_{i \in \mathcal{U}_{ID}} u_i &= g_1^{J_u(ID)} g^{K_u(ID)},\\
v' \prod_{j \in \mathcal{M}} v_j &= g_1^{J_m(m)} g^{K_m(m)}.
\end{aligned}
$$

Also note that by these assignments, $\mathcal{B}$ does not know the master secret key, $msk = g^{a^2}$, and he must simulate $\mathcal{C}$ and answer all $\mathcal{A}_I$'s queries in Game I without the knowledge of msk.

Phase 1 queries: In this step, $\mathcal{A}_I$ has access to $\mathcal{O}_I = \{\mathcal{O}_{pk}, \mathcal{O}_{psk}, \mathcal{O}_{Replace.pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$. $\mathcal{B}$ responds to $\mathcal{A}_I$'s queries by simulating these oracles as follows:

- $\mathcal{O}_{pk}$. As $\mathcal{A}_I$ sends a public key query for an identity ID to $\mathcal{O}_{pk}$, $\mathcal{B}$ checks whether such key exists in the database $\mathcal{DB}$. If so, $\mathcal{B}$ returns this public key to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ runs the User-Key-Generation algorithm to generate pk_{ID} and returns it to $\mathcal{A}_I$. Also $\mathcal{B}$ adds pk_{ID} and its corresponding x_{ID} in the database.
- $\mathcal{O}_{psk}$. As $\mathcal{A}_I$ sends a partial private key query for an identity ID to $\mathcal{O}_{psk}$, $\mathcal{B}$ checks whether such key exists in the database $\mathcal{DB}$. If so, $\mathcal{B}$ returns this partial private key to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ tries to generate psk_{ID} without the knowledge of the master secret key as follows:
  - If $J_u(ID) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
  - If $J_u(ID) \neq 0 \bmod q$, $\mathcal{B}$ randomly selects $r \in Z_q^*$ and creates psk_{ID} as follows:

$$
\begin{aligned}
psk_{ID} &= (psk_{ID,1}, psk_{ID,2})\\
&= (g_1^{-\frac{K_u(ID)}{J_u(ID)}} (u' \prod_{i \in \mathcal{U}_{ID}} u_i)^r, g_1^{-\frac{1}{J_u(ID)}} g^r).
\end{aligned}
$$

  Then $\mathcal{B}$ returns psk_{ID} to $\mathcal{A}_I$ and also adds it in the database.

  By defining $\tilde{r} = r - a/J_u(ID)$, it is easy to check that $g_1^{-K_u(ID)/J_u(ID)} (u' \prod_{i \in \mathcal{U}_{ID}} u_i)^r = g^{\alpha^2} (u' \prod_{i \in \mathcal{U}_{ID}} u_i)^{\tilde{r}}$ and $g_1^{-1/J_u(ID)} g^r = g^{\tilde{r}}$, and as a result, psk_{ID} which is simulated by $\mathcal{B}$, has the correct construction and from the perspective of $\mathcal{A}_I$, all the partial private keys generated by $\mathcal{B}$ are indistinguishable from those created by the true challenger $\mathcal{C}$.
- $\mathcal{O}_{Replace.pk}$. Suppose that $\mathcal{A}_I$ requests to replace the public key of an identity ID, i.e., pk_{ID} corresponding to x_{ID}, with a new public key $pk'_{ID} = (pk'_{ID,1}, pk'_{ID,2})$, corresponding to x'_{ID}. $\mathcal{B}$ firstly checks whether $e(pk'_{ID,1}, pk'_{ID,2}) = T$. If so, $\mathcal{B}$ replaces the (x_{ID}, pk_{ID}) with (x'_{ID}, pk'_{ID}) in the database. If there is not any (x_{ID}, pk_{ID}) corresponding to the identity ID, $\mathcal{B}$ directly sets $(x_{ID}, pk_{ID}) = (x'_{ID}, pk'_{ID})$ in the database. After the replacement, $\mathcal{B}$ sets $sta = 1$ for the identity ID.

- $\mathcal{O}_{sk}$. As $\mathcal{A}_I$ sends a private key query for an identity ID to $\mathcal{O}_{sk}$, $\mathcal{B}$ checks whether such key exists in the database $\mathcal{DB}$. If so, $\mathcal{B}$ returns this private key to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ searches $\mathcal{DB}$ for psk_{ID}. If psk_{ID} exists in $\mathcal{DB}$, $\mathcal{B}$ picks it, otherwise $\mathcal{B}$ acts as follows:
  - If $J_u(ID) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
  - If $J_u(ID) \neq 0 \bmod q$, $\mathcal{B}$ creates psk_{ID} similar to that in simulating $\mathcal{O}_{psk}$.

  Then $\mathcal{B}$ obtains (x_{ID}, pk_{ID}). ($\mathcal{B}$ picks (x_{ID}, pk_{ID}) from the database if exists and the corresponding $sta = 0$, otherwise $\mathcal{B}$ creates (x_{ID}, pk_{ID}) by running the User-Key-Generation algorithm). Afterwards, $\mathcal{B}$ can produce sk_{ID} by running the Private-Key-Generation algorithm, since he knows both x_{ID} and psk_{ID}. So $\mathcal{B}$ generates sk_{ID}, returns it to $\mathcal{A}_I$ and also adds it in the database.
- $\mathcal{O}_{Signcrypt}$. As $\mathcal{A}_I$ sends a signcryption query for (M, ID_S, ID_R), $\mathcal{B}$ gets the private key of the signer sk_{ID_S} by simulating $\mathcal{O}_{sk}$ as mentioned and creates a signcryption $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$ on message M by running the Signcryption algorithm. Then $\mathcal{B}$ returns σ to $\mathcal{A}_I$. If $\mathcal{B}$ cannot simulate sk_{ID_S} (i.e., $J_u(ID_S) = 0 \bmod q$), $\mathcal{B}$ aborts the simulation.
- $\mathcal{O}_{Unsigncrypt}$. As $\mathcal{A}_I$ sends an unsigncrypt query for $(\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5), ID_S, ID_R)$, $\mathcal{B}$ firstly runs the verification part of the Unsigncryption algorithm. If the verification fails, $\mathcal{B}$ returns $\perp$ to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ extracts M as follows:
  - If $sta = 0$ for ID_R (i.e., pk_{ID_R} has never been replaced), $\mathcal{B}$ checks whether sk_{ID_R} exists in the database $\mathcal{DB}$. If so, picks it. Otherwise, $\mathcal{B}$ obtains sk_{ID_R} by simulating $\mathcal{O}_{sk}$ (assume that $J_u(ID_R) \neq 0 \bmod q$). Then $\mathcal{B}$ executes the unsigncrypt part of the Unsigncryption algorithm to obtain M and returns it to $\mathcal{A}_I$.
  - If $sta = 1$ for ID_R (i.e., pk_{ID_R} has been replaced), $\mathcal{B}$ acts as follows:
    - If $J_u(ID) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
    - If $J_u(ID) \neq 0 \bmod q$, $\mathcal{B}$ firstly computes $g_1^{r_1}$ as follows:

$$g_1^{r_1} = \left(\frac{\sigma_3}{\sigma_2^{K_u(ID_R)}}\right)^{\frac{1}{J_u(ID_R)}}.$$

Note that:

$$\begin{aligned}
&\left(\frac{\sigma_3}{\sigma_2^{K_u(ID_R)}}\right)^{\frac{1}{J_u(ID_R)}} \\
&= \left(\frac{(u' \prod_{i \in \mathcal{U}_{ID_R}} u_i)^{r_1}}{g^{r_1 . K_u(ID_R)}}\right)^{\frac{1}{J_u(ID_R)}} \\
&= \left(\frac{(g_1^{J_u(ID_R)} g^{K_u(ID_R)})^{r_1}}{g^{r_1 . K_u(ID_R)}}\right)^{\frac{1}{J_u(ID_R)}} \\
&= g_1^{r_1}.
\end{aligned}$$

Then $\mathcal{B}$ retrieves the x_{ID_R} corresponding to pk_{ID_R} from the database or gets it from $\mathcal{A}_I$ (Note that pk_{ID_R} is a replaced public key, since $sta = 1$ and so $\mathcal{A}_I$ knows it). Afterwards, with the knowledge of $g_1^{r_1}$ and x_{ID_R}, $\mathcal{B}$ can extract M as follows:

$$M = \frac{\sigma_1}{e(g_1^{r_1}, g_1^{x_{ID_R}^2})}.$$

So, $\mathcal{B}$ computes M as above and returns it to $\mathcal{A}_I$.

Challenge: After a polynomially bounded number of queries from $\mathcal{O}_I$, $\mathcal{A}_I$ selects two distinct identities ID_{S^*} and ID_{R^*} and two equal length messages $M_0, M_1 \in G_2$ as her challenge. (Note that $\mathcal{A}_I$ has never issued a private key query for ID_{R^*} from $\mathcal{O}_{sk}$.) Then $\mathcal{A}_I$ submits $\{ID_{S^*}, ID_{R^*}\}$ and $\{M_0, M_1\}$ to the challenger $\mathcal{C}$. $\mathcal{B}$ plays the role of $\mathcal{C}$ as follows:

- If $J_u(ID_{R^*}) \neq 0 \bmod q$ or $J_u(ID_{S^*}) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
- If $J_u(ID_{R^*}) = 0 \bmod q$ and $J_u(ID_{S^*}) \neq 0 \bmod q$, $\mathcal{B}$ selects a bit γ by flipping a fair coin and creates a signcryption on M_γ for the challenge identities ID_{S^*} and ID_{R^*} as follows.

  Let $pk_{ID_{S^*}} = (g_1^{x_{ID_{S^*}}}, g_1^{1/x_{ID_{S^*}}})$ and $pk_{ID_{R^*}} = (g_1^{x_{ID_{R^*}}}, g_1^{1/x_{ID_{R^*}}})$ be the current public keys of ID_{S^*} and ID_{R^*}, respectively. $\mathcal{B}$ firstly retrieves $x_{ID_{S^*}}$ and $x_{ID_{R^*}}$. ($\mathcal{B}$ can retrieve these values from $\mathcal{DB}$ if exist, otherwise $\mathcal{B}$ retrieves them by running the User-Key-Generation algorithm). Then $\mathcal{B}$ sets the followings: (Note that $(g, A = g^a, X)$ is the input of the $(\mathcal{S}_1, 3)$-BDHE-Set problem which $\mathcal{B}$ is trying to solve it.)

$$\begin{aligned}
\sigma_1^* &= X^{x_{ID_{R^*}}^2} M_\gamma, \\
\sigma_2^* &= A = g_1, \\
\sigma_3^* &= A^{K_u(ID_{R^*})}, \\
\sigma_4^* &= (A^{x_{ID_{S^*}}^2})^{\frac{-1}{J_u(ID_{S^*})}} g^{t^*}, \quad t^* \in_R Z_q^*.
\end{aligned}$$

Let $m_\gamma = H_1(\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, ID_R^*, pk_{ID_R^*,1})$ and $\mathcal{M}_\gamma = \{j | m_\gamma[j] = 1, j = 1, 2, \ldots, n_m\}$, where $m_\gamma[j]$ denotes the j-th bit of $m_\gamma \in \{0,1\}^{n_m}$. Also, let $h^* = H_2(ID_{S^*}, m_\gamma, pk_{ID_{S^*},1}, \sigma_4^*, \sigma_2^*)$.
- If $J_m(m_\gamma) + x_{ID_{S^*}} h^* \neq 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
- If $J_m(m_\gamma) + x_{ID_{S^*}} h^* = 0 \bmod q$, $\mathcal{B}$ sets

$$\sigma_5^* = (A^{x_{ID_{S^*}}^2})^{\frac{-K_u(ID_{S^*})}{J_u(ID_{S^*})}} (u' \prod_{i \in \mathcal{U}_{ID_S^*}} u_i)^{t^*} A^{K_m(m_\gamma)},$$

and gives $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, \sigma_5^*)$ to $\mathcal{A}_I$. It is easy to check that if $X = e(g,g)^{a^3}$ (i.e., $(g, A = g^a, X)$ is a valid $(\mathcal{S}_1, 3)$-BDHE-Set tuple), σ^* is a valid signcryption on M_γ by assigning α, r, r', r_1 and r_2 in the proposed scheme as follows:

- $r_1 = a$,
- $\alpha = a$ (and as a result $msk = g^{a^2}$ and $g_1 = g^a = A$),
- $r' + r_2 = t^*$,
- $r = -a/J_u(ID_{S^*})$.

Otherwise, (if X is a random element of G_2 and not equal to $e(g,g)^{a^3}$), σ^* is a random string which is not a valid signcryption neither for M_0 nor for M_1.

Phase 2 queries: $\mathcal{A}_I$ continuous to her queries from $\mathcal{O}_I$ and $\mathcal{B}$ responds to these queries similar to those in Phase 1 queries. Note that $\mathcal{A}_I$ is not allowed to send an unsigncryption query on σ^* under ID_{S^*} and ID_{R^*} unless $pk_{ID_{S^*}}$ or $pk_{ID_{R^*}}$ used to signcrypt M_γ, has been replaced after the challenge was issued.

Response: At the end of the simulation, $\mathcal{A}_I$ outputs a guess γ^* of γ. Finally, when the Game I between $\mathcal{A}_I$ and $\mathcal{B}$ terminates, $\mathcal{B}$ acts as follows:

- If the simulation fails in any step (because of aborting $\mathcal{B}$) $\mathcal{B}$ randomly selects its guess β' of β.
- Otherwise, if $\gamma^* = \gamma$, $\mathcal{B}$ outputs a guess $\beta' = 1$, implying that $X = e(g,g)^{a^3}$, else $\mathcal{B}$ outputs $\beta' = 0$ to the $(\mathcal{S}_1, 3)$-BDHE-Set problem.

Time Analysis: Noting the above descriptions we can see that $\mathcal{B}$ needs a time $t' \leq t + t_I$, for running the game.

Probability Analysis: Define the success probability of $\mathcal{B}$ in solving the $(\mathcal{S}_1, 3)$-BDHE-Set problem as $Pr[\mathcal{B}\ wins]$ and the success probability of $\mathcal{A}_I$ in Game I as $Pr[\mathcal{A}_I\ wins]$. Noting that if $\mathcal{B}$ aborts $Pr[\mathcal{B}\ wins] = \frac{1}{2}$ (since $\mathcal{B}$ randomly selects its guess β' of β.), and assuming that $Pr[\mathcal{A}_I\ wins] \geq \frac{1}{2} + \varepsilon$, we have:

$$\begin{aligned} Pr[\mathcal{B}\ wins] &= Pr[\mathcal{B}\ wins|abort]Pr[abort] \\ &\quad + Pr[\mathcal{B}\ wins|\overline{abort}]Pr[\overline{abort}] \\ &= \frac{1}{2}Pr[abort] + Pr[\mathcal{A}_I\ wins]Pr[\overline{abort}] \\ &\geq \frac{1}{2}(1 - Pr[\overline{abort}]) + (\frac{1}{2} + \varepsilon)Pr[\overline{abort}] \\ &= \frac{1}{2} + \varepsilon.Pr[\overline{abort}]. \end{aligned}$$

$\mathcal{B}$ will not abort if all the following independent events happen:

- E_1: $J_u(ID_{S^*}) \neq 0 \bmod q$, and $J_u(ID) \neq 0 \bmod q$ for all queries from $\mathcal{O}_{psk}$, $\mathcal{O}_{sk}$, $\mathcal{O}_{Signcrypt}$ and $\mathcal{O}_{Unsigncrypt}$. Let $E_{1,0}$ denotes the event that $J_u(ID_{S^*}) \neq 0 \bmod q$, and $E_{1,i}$ denotes the event that $J_u(ID) \neq 0 \bmod q$ in the i-th query from the mentioned oracles, hence $E_1 = \bigcap_{i=0}^{q_{psk}+q_{sk}+q_S+q_U} E_{1,i}$.
- E_2: $J_u(ID_{R^*}) = 0 \bmod q$.
- E_3: $J_m(m_\gamma) + x_{ID_{S^*}} h^* = 0 \bmod q$.

It is easy to see that [5]:

$$Pr[J_u(ID) = 0 \bmod q] = \frac{1}{l_u(n_u + 1)},$$

and [17]:

$$Pr[J_m(m_\gamma) + x_{ID_{S^*}} h^* \bmod q] = \frac{1}{l_m(n_m + 1)}.$$

As a result:

$$\begin{aligned} Pr[\overline{abort}] &\geq Pr[E_1 \bigcap E_2 \bigcap E_3] = Pr[E_1].Pr[E_2].Pr[E_3] \\ &= Pr[\bigcap_{i=0}^{q_{psk}+q_{sk}+q_S+q_U} E_{1,i}].Pr[E_2].Pr[E_3] \\ &\geq (1 - \frac{q_{psk} + q_{sk} + q_S + q_U + 1}{l_u(n_u + 1)}).\frac{1}{l_u(n_u + 1)l_m(n_m + 1)} \\ &\geq (1 - \frac{q_{psk} + q_{sk} + q_S + q_U + 1}{l_u}).\frac{1}{l_u(n_u + 1)l_m(n_m + 1)} \\ &= \frac{1}{8q_U(q_{psk} + q_{sk} + q_S + q_U + 1)(n_u + 1)(n_m + 1)} = \varepsilon_I, \end{aligned}$$

where the equality before the rightmost one is implied from $l_u = 2(q_{psk} + q_{sk} + q_S + q_U + 1)$ and $l_m = 2q_U$. Finally we have:

$$Pr[\mathcal{B}\ wins] \geq \frac{1}{2} + \varepsilon.\varepsilon_I.$$

As the final result, if $\mathcal{A}_I$ can win Game I with a non-negligible advantage ε (i.e., guess γ correctly with probability at least $\frac{1}{2} + \varepsilon$ for a non-negligible value of ε), then $\mathcal{B}$ can solve an instance of the $(\mathcal{S}_1, 3)$-BDHE-Set problem with a non-negligible advantage ε' (i.e., guess β correctly with probability at least $\frac{1}{2} + \varepsilon'$), where $\varepsilon' \geq \varepsilon.\varepsilon_I$ and this is a contradiction of the $(\mathcal{S}_1, 3)$-BDHE-Set assumption in complexity theory. ■

Lemma 2. The proposed scheme is $(\varepsilon, t, q_{pk}, 0, 0, q_sk, q_S, q_U)$-semantically secure against $\mathcal{A}_{II}$, if the (ε', t')-$(\mathcal{S}_2, 5)$-BDHE-Set assumption holds in (G_1, G_2), where $\varepsilon' \geq \varepsilon\varepsilon_{II}$ and $t' \leq t + t_{II}$.

Proof. Suppose that there exists a $(\varepsilon, t, q_{pk}, 0, 0, q_sk, q_S, q_U)$-type II adversary $\mathcal{A}_{II}$, who can break the indistinguishability of encryptions against adaptive chosen ciphertext attack (IND-CCA) in the proposed scheme according to Game II. By this assumption, we can construct a simulator $\mathcal{B}$ that can use $\mathcal{A}_{II}$ as a subroutine and solve the $(\mathcal{S}_2, 5)$-BDHE-Set problem with a probability at least ε' and in time at most t', which contradicts the (ε', t')-$(\mathcal{S}_2, 5)$-BDHE-Set assumption in (G_1, G_2).

Consider two multiplicative cyclic groups G_1 and G_2 of a large prime order q and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$. Suppose that $\mathcal{B}$ is given a random $(\mathcal{S}_2, 5)$-BDHE-Set challenge $(A = h \in G_1, B = h^a \in G_1, C = h^{a^2} \in G_1, X \in G_2)$ and outputs a guess $\beta' = 1$, if he decides that $X = e(h,h)^{a^5}$ and $\beta' = 0$, otherwise. Also suppose that $B = h^a$ is a generator of G_1. In

order to use $\mathcal{A}_{II}$ as a subroutine, $\mathcal{B}$ must simulate the challenger $\mathcal{C}$ and answer all $\mathcal{A}_{II}$'s queries in Game II. In order to respond these queries consistently, $\mathcal{B}$ creates a database $\mathcal{DB} = \{(ID, x_{ID}, pk_{ID}, sk_{ID})\}$ which is initially empty. Then $\mathcal{B}$ plays Game II with $\mathcal{A}_{II}$ and simulates $\mathcal{C}$ and all oracles which $\mathcal{A}_{II}$ has access to them, i.e., $\mathcal{O}_{II} = \{\mathcal{O}_{pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unigncrypt}\}$, as follows:

Initialization: Let $l_u = 2(q_{sk} + q_S + q_U + 1)$ and $l_m = 2q_U$. Assume that $l_u(n_u + 1) < q$ and $l_m(n_m + 1) < q$. $\mathcal{A}_{II}$ selects a random $\alpha \in Z_q^*$ and sets $g_1 = g^\alpha$. Then $\mathcal{A}_{II}$ selects the values k_u, $k_m, x', x_1, \ldots, x_{n_u}, y', y_1, \ldots, y_{n_u}, z', z_1, \ldots, z_{n_m}$ and $w', w_1, \ldots, w_{n_m}$ similar to them in the proof of Lemma 1 and assigns:

$$\begin{aligned} g &= B = h^a, \\ g_1 &= B^\alpha = g^\alpha, \\ u' &= C^{-k_u l_u + x'} B^{y'}, \\ u_i &= C^{x_i} B^{y_i} \quad (i = 1, 2, \ldots, n_u), \\ v' &= C^{-k_m l_m + z'} B^{w'}, \\ v_j &= C^{z_j} B^{w_j} \quad (j = 1, 2, \ldots, n_m). \end{aligned}$$

$\mathcal{A}_{II}$ also computes $T = e(g_1, g_1)$ and selects two collision resistant hash functions $H_1 : \{0,1\}^* \longrightarrow \{0,1\}^{n_m}$ and $H_2 : \{0,1\}^* \longrightarrow Z_q^*$ and sends $params = \{G_1, G_2, q, e, g, g_1, T, u', v', U, V, H_1, H_2\}$ and α to $\mathcal{B}$. $\mathcal{A}_{II}$ also computes four functions $J_u(ID)$, $K_u(ID)$, $J_m(m)$ and $K_m(m)$ similar to those in the proof of Lemma 1 and sends them to $\mathcal{B}$.

By these assignments, the following equations hold:

$$u' \prod_{i \in \mathcal{U}_{ID}} u_i = C^{J_u(ID)} B^{K_u(ID)},$$
$$v' \prod_{j \in \mathcal{M}} v_j = C^{J_m(m)} B^{K_m(m)}.$$

Also, note that in Game II (in contrast to Game I), $\mathcal{B}$ knows the master secret key, $msk = g^{\alpha^2}$, and it must simulate $\mathcal{C}$ and answer all $\mathcal{A}_{II}$'s queries by this fact.

Phase 1 queries: In this step, $\mathcal{A}_{II}$ has access to $\mathcal{O}_{II} = \{\mathcal{O}_{pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$. (Note that $\mathcal{A}_{II}$ herself possesses the msk and can generate psk, so she does not need to has queries from $\mathcal{O}_{psk}$). $\mathcal{B}$ responds to $\mathcal{A}_{II}$' s queries by simulating the oracles in $\mathcal{O}_{II}$ as follows:

- $\mathcal{O}_{pk}$. As $\mathcal{A}_{II}$ sends a public key query for an identity ID to $\mathcal{O}_{pk}$, $\mathcal{B}$ checks whether such key exists in the database $\mathcal{DB}$. If so, $\mathcal{B}$ returns this public key to $\mathcal{A}_{II}$. Otherwise, $\mathcal{B}$ chooses a random $x_{ID} \in_R Z_q^*$ and sets $pk_{ID} = (pk_{ID,1}, pk_{ID,2}) = (C^{\alpha x_{ID}}, A^{\alpha / x_{ID}})$ and returns it to $\mathcal{A}_{II}$. Also $\mathcal{B}$ adds (x_{ID}, pk_{ID}) in the database. Note that by this assignment, the real secret value of an identity ID is equal to ax_{ID} and since $\mathcal{B}$ does not know a, it does not know the real secret value. So, $\mathcal{B}$ must answer the following queries of $\mathcal{A}_{II}$ without the knowledge of the real secret value of an identity ID.
- $\mathcal{O}_{sk}$. As $\mathcal{A}_{II}$ sends a private key query for an identity ID to $\mathcal{O}_{sk}$, $\mathcal{B}$ checks whether such key exists in the database $\mathcal{DB}$. If so, $\mathcal{B}$ returns this private key to $\mathcal{A}_{II}$. Otherwise $\mathcal{B}$ acts as follows:
  - If $J_u(ID) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
  - If $J_u(ID) \neq 0 \bmod q$, $\mathcal{B}$ checks whether (x_{ID}, pk_{ID}) exists in the database. If so, $\mathcal{B}$ picks it, otherwise $\mathcal{B}$ produces (x_{ID}, pk_{ID}) by simulating $\mathcal{O}_{pk}$. Then $\mathcal{B}$ selects a random $r \in_R Z_q^*$ and assigns the private key as:

$$\begin{aligned} sk_{ID} =& (sk_{ID,1}, sk_{ID,2}) \\ =& (C^{-\frac{K_u(ID)}{J_u(ID)}(\alpha x_{ID})^2} (u' \prod_{i \in \mathcal{U}_{ID}} u_i)^r, \\ & C^{-\frac{(\alpha x_{ID})^2}{J_u(ID)}} B^r). \end{aligned}$$

  Finally, $\mathcal{B}$ returns sk_{ID} to $\mathcal{A}_{II}$ and also adds it in the database. Note that By defining $\tilde{r} = r - a(\alpha x_{ID})^2 / J_u(ID)$, it is easy to see that in the above assignments, $sk_{ID,1} = g^{(a\alpha x_{ID})^2} (u' \prod_{i \in \mathcal{U}_{ID}} u_i)^{\tilde{r}}$, and $sk_{ID,2} = g^{\tilde{r}}$, and as a result by assuming ax_{ID} as the secret value of identity ID, sk_{ID} which is simulated by $\mathcal{B}$, has the correct construction and from the perspective of $\mathcal{A}_{II}$, all the private keys generated by $\mathcal{B}$ are indistinguishable from those created by the true challenger $\mathcal{C}$.
- $\mathcal{O}_{Signcrypt}$. As $\mathcal{A}_{II}$ sends a signcrypt query for (M, ID_S, ID_R), $\mathcal{B}$ gets the private key of the signer sk_{ID_S} by simulating $\mathcal{O}_{sk}$ as mentioned and creates a signcryption $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$ on message M by running the Signcryption algorithm. Then $\mathcal{B}$ returns σ to $\mathcal{A}_{II}$. If $\mathcal{B}$ cannot simulate sk_{ID_S} (i.e., $J_u(ID_S) = 0 \bmod q$), $\mathcal{B}$ aborts the simulation.
- $\mathcal{O}_{Unsigncrypt}$. As $\mathcal{A}_{II}$ sends an unsigncrypt query for $(\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5), ID_S, ID_R)$, $\mathcal{B}$ firstly runs the verification part of the Unsigncryption algorithm. If the verification fails, $\mathcal{B}$ returns $\perp$ to $\mathcal{A}_{II}$. Otherwise, $\mathcal{B}$ checks whether sk_{ID_R} exists in the database $\mathcal{DB}$. If so, picks it. Otherwise, $\mathcal{B}$ obtains sk_{ID_R} by simulating $\mathcal{O}_{sk}$ (assume that $J_u(ID_R) \neq 0 \bmod q$). Then $\mathcal{B}$ executes the Unsigncrypt part of the Unsigncryption algorithm to obtain M and returns it to $\mathcal{A}_{II}$.

Challenge: After a polynomially bounded number of queries from $\mathcal{O}_{II}$, $\mathcal{A}_{II}$ selects two distinct identities ID_{S^*} and ID_{R^*} and two equal length messages $M_0, M_1 \in G_2$ as her challenge. Note that $\mathcal{A}_{II}$ has never issued a private key query for ID_{R^*} from $\mathcal{O}_{sk}$.

Then $\mathcal{A}_{II}$ submits (ID_{S^*}, ID_{R^*}) and $\{M_0, M_1\}$ to the challenger $\mathcal{C}$. $\mathcal{B}$ plays the role of $\mathcal{C}$ as follows:

- If $J_u(ID_{R^*}) \neq 0 \bmod q$ or $J_u(ID_{S^*}) = 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
- If $J_u(ID_{R^*}) = 0 \bmod q$ and $J_u(ID_{S^*}) \neq 0 \bmod q$, $\mathcal{B}$ selects a bit γ by flipping a fair coin and creates a signcryption on M_γ for the challenge identities ID_{S^*} and ID_{R^*} as follows:

  Let $pk_{ID_{S^*}} = (C^{\alpha x_{ID_{S^*}}}, A^{\alpha/x_{ID_{S^*}}})$ and $pk_{ID_{R^*}} = (C^{\alpha x_{ID_{R^*}}}, A^{\alpha/x_{ID_{R^*}}})$ be the current public keys of ID_{S^*} and ID_{R^*}, respectively. $\mathcal{B}$ firstly retrieves $x_{ID_{S^*}}$ and $x_{ID_{R^*}}$. ($\mathcal{B}$ can retrieve these values from $\mathcal{DB}$ if exist, otherwise $\mathcal{B}$ retrieves them by simulating $\mathcal{O}_{pk}$). Then $\mathcal{B}$ sets the followings: (Note that $(A = h, B = h^a, C = h^{a^2}, X)$ is the input of $(\mathcal{S}_2, 5)$-BDHE-Set problem which $\mathcal{B}$ is trying to solve it.)

$$\sigma_1^* = X^{(\alpha x_{ID_{R^*}})^2} M_\gamma,$$
$$\sigma_2^* = C = h^{a^2},$$
$$\sigma_3^* = C^{K_u(ID_{R^*})},$$
$$\sigma_4^* = (C^{(\alpha x_{ID_{S^*}})^2})^{\frac{-1}{J_u(ID_{S^*})}} g^{t^*}, \quad t^* \in_R Z_q^*.$$

  Let $m_\gamma = H_1(\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, ID_R^*, pk_{ID_R^*,1})$ and $\mathcal{M}_\gamma = \{j | m_\gamma[j] = 1, j = 1, 2, \ldots, n_m\}$, where $m_\gamma[j]$ denotes the j-th bit of $m_\gamma \in \{0,1\}^{n_m}$. Also, let $h^* = H_2(ID_{S^*}, m_\gamma, pk_{ID_{S^*},1}, \sigma_4^*, \sigma_2^*)$.
  - If $J_m(m_\gamma) + \alpha x_{ID_{S^*}} h^* \neq 0 \bmod q$, $\mathcal{B}$ aborts the simulation.
  - If $J_m(m_\gamma) + \alpha x_{ID_{S^*}} h^* = 0 \bmod q$, $\mathcal{B}$ sets

$$\sigma_5^* = (C^{(\alpha x_{ID_{S^*}})^2})^{\frac{-K_u(ID_{S^*})}{J_u(ID_{S^*})}} (u' \prod_{i \in \mathcal{U}_{ID_S^*}} u_i)^{t^*} C^{K_m(m_\gamma)},$$

  and gives $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, \sigma_5^*)$ to $\mathcal{A}_{II}$. It is easy to check that if $X = e(h,h)^{a^5}$ (i.e., $A = h, B = h^a, C = h^{a^2}, X$ is a valid $(\mathcal{S}_2, 5)$-BDHE-Set tuple), σ^* is a valid signcryption on M_γ by assigning r, r', r_1 and r_2 in the proposed scheme as follows:
  - $r_1 = a$,
  - $r' + r_2 = t^*$,
  - $r = -\alpha^2 / a J_u(ID_{S^*})$.

  Otherwise, (if X is a random element of G_2 and not equal to $e(h,h)^{a^5}$), σ^* is a random string which is not a valid signcryption neither for M_0 nor for M_1.

Phase 2 queries: $\mathcal{A}_{II}$ continuous to her queries from $\mathcal{O}_{II}$ and $\mathcal{B}$ responds to these queries similar to that in Phase 1 queries. Note that $\mathcal{A}_{II}$ is not allowed to send an unsigncryption query on σ^* under ID_S^* and ID_R^*.

Response: At the end of the simulation, $\mathcal{A}_{II}$ outputs a guess γ^* of γ. Finally, when the Game II between $\mathcal{A}_{II}$ and $\mathcal{B}$ terminates, $\mathcal{B}$ acts as follows:

- If the simulation fails in any step (because of aborting $\mathcal{B}$) $\mathcal{B}$ randomly selects its guess β' of β.
- Otherwise, If $\gamma^* = \gamma$, $\mathcal{B}$ outputs a guess $\beta' = 1$, implying that $X = e(h,h)^{a^5}$, else $\mathcal{B}$ outputs $\beta' = 0$ to the $(\mathcal{S}_2, 5)$-BDHE-Set problem.

Time and probability analyses are similar to those in the proof of Lemma 1 except for $q_{psk} = 0$. ■

Theorem 1. The proposed scheme is semantically secure under adaptive chosen ciphertext attack (according to Definition 1) in the standard model under the $(\mathcal{S}_2, 5)$-BDHE-Set assumption.

Proof. The proof is directly implied from Lemma 1 and Lemma 2. ■

Lemma 3. The proposed scheme is $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$-unforgeable against $\mathcal{A}_I$, if the (ε', t')-2-CDHE assumption holds in G_1, where $\varepsilon' \geq \varepsilon\varepsilon_I$ and $t' \leq t + t_I$.

Proof. Suppose that there exists a $(\varepsilon, t, q_{pk}, q_{psk}, q_{Rpk}, q_{sk}, q_S, q_U)$- type I adversary $\mathcal{A}_I$, who can break the unforgeability against adaptive chosen message attack (EUF-CMA) in the proposed scheme according to Game III. By this assumption, we can construct a simulator $\mathcal{B}$ that can use $\mathcal{A}_I$ as a subroutine and solve an instance of a 2-CDHE problem with a probability at least ε' and in time at most t', which contradicts the (ε', t')-2-CDHE assumption in G_1.

Consider a multiplicative cyclic groups G_1 of a large prime order q and a random generator g of G_1. Suppose that $\mathcal{B}$ is given a random 2-CDHE challenge $(g \in G_1, A = g^a \in G_1)$ and is requested to output $g^{a^2} \in G_1$. In order to use $\mathcal{A}_I$ as a subroutine, $\mathcal{B}$ must simulate the challenger $\mathcal{C}$ and answer all $\mathcal{A}_I$'s queries in Game III. $\mathcal{B}$ firstly selects a group G_2 of order q and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$. Then $\mathcal{B}$ plays Game III with $\mathcal{A}_I$ and simulates $\mathcal{C}$ and all oracles which $\mathcal{A}_I$ has access to them, i.e., $\mathcal{O}_I = \{\mathcal{O}_{pk}, \mathcal{O}_{psk}, \mathcal{O}_{Replace.pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$, as follows:

Initialization: It is similar to the Initialization step of the proof of Lemma 1.

Queries: $\mathcal{B}$ responds all $\mathcal{A}_I$'s queries from $\mathcal{O}_I$ similar to those in the Phase 1 queries of the proof of Lemma 1.

Output: After a polynomially bounded number of queries (if $\mathcal{B}$ does not abort), $\mathcal{A}_I$ outputs a new valid signcryption $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, \sigma_5^*)$ on message M^* for ID_{S^*} with public key $pk_{ID_{S^*}} = (g_1^{x_{ID_{S^*}}}, g_1^{1/x_{ID_{S^*}}})$ and ID_{R^*} with public key $pk_{ID_{R^*}} = (g_1^{x_{ID_{R^*}}}, g_1^{1/x_{ID_{R^*}}})$. (Note that $(\sigma^*,$

ID_{S^*}, ID_{R^*}) is not produced by the signcryption oracle.) Finally, when the Game III between $\mathcal{A}_I$ and $\mathcal{B}$ terminates, $\mathcal{B}$ computes $m^* = H_1(\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, ID_{R^*}, pk_{ID_{R^*}})$ and acts as follows:

- If $J_u(ID_{S^*}) \neq 0 \bmod q$ or $J_m(m^*) + x_{ID_{S^*}} h^* \neq 0 \bmod q$ (or the simulation fails in any steps), $\mathcal{B}$ aborts.
- If $J_u(ID_{S^*}) = 0 \bmod q$ and $J_m(m^*) + x_{ID_{S^*}} h^* = 0 \bmod q$ (and the simulation does not fail in any steps), $\mathcal{B}$ retrieves $x_{ID_{S^*}}$ and computes:

$$g^{a^2} = \left(\frac{\sigma_5^*}{(\sigma_4^*)^{K_u(ID_{S^*})}(\sigma_2^*)^{K_m(m^*)}}\right)^{\frac{1}{x^2_{ID_{S^*}}}}.$$

Time Analysis: Noting the above descriptions we can see that $\mathcal{B}$ needs a time $t' \leq t + t_I$, for running the game.

Probability Analysis: Define the success probability of $\mathcal{B}$ in solving the 2-CDHE problem as $Pr[\mathcal{B}\ wins]$ and the success probability of $\mathcal{A}_I$ in Game III as $Pr[\mathcal{A}_I\ wins]$. Assuming that $Pr[A_I\ wins] \geq \varepsilon$, we have:

$$\begin{aligned} Pr[\mathcal{B}\ wins] =& Pr[\overline{abort} \bigcap \mathcal{A}_I\ wins] \\ =& Pr[\mathcal{A}_I\ wins].Pr[\overline{abort}] \\ \geq& \varepsilon.Pr[\overline{abort}] \geq \varepsilon\varepsilon_I, \end{aligned}$$

since $Pr[\overline{abort}] \geq \varepsilon_I$, as is proved in the probability analysis of Lemma 1.

As a result, if $\mathcal{A}_I$ can win Game III with a non-negligible advantage ε (i.e., forge a valid signcryption with probability at least ε for a non-negligible value of ε), then $\mathcal{B}$ can solve an instance of the 2-CDHE problem with a non-negligible probability ε', where $\varepsilon' \geq \varepsilon\varepsilon_I$ and this is a contradiction of the 2-CDHE assumption in complexity theory. ■

Lemma 4. The proposed scheme is $(\varepsilon, t, q_{pk}, 0, 0, q_{sk}, q_S, q_U)$-unforgeable against $\mathcal{A}_{II}$, if the (ε', t')-3-CDHE assumption holds in G_1, where $\varepsilon' \geq \varepsilon\varepsilon_{II}$ and $t' \leq t + t_{II}$.

Proof. Suppose that there exists a $(\varepsilon, t, q_{pk}, 0, 0, q_{sk}, q_S, q_U)$-type II adversary $\mathcal{A}_{II}$, who can break the unforgeability against adaptive chosen message attack (EUF-CMA) in the proposed scheme according to Game IV. By this assumption, we can construct a simulator $\mathcal{B}$ that can use $\mathcal{A}_{II}$ as a sub-routine and solve an instance of a 3-CDHE problem with a probability at least ε' and in time at most t', which contradicts the (ε', t')-3-CDHE assumption in G_1.

Consider a multiplicative cyclic groups G_1 of a large prime order q and a random generator $B = h^a$ of G_1. Suppose that $\mathcal{B}$ is given a random 3-CDHE challenge $(A = h \in G_1, B = h^a \in G_1, C = h^{a^2} \in G_1)$ and is requested to output $h^{a^3} \in G_1$. In order to use $\mathcal{A}_{II}$ as a subroutine, $\mathcal{B}$ must simulate the challenger $\mathcal{C}$ and answer all $\mathcal{A}_{II}$'s queries in Game IV. $\mathcal{B}$ firstly selects a group G_2 of order q and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$. Then $\mathcal{B}$ plays Game IV with $\mathcal{A}_{II}$ and simulates $\mathcal{C}$ and all oracles which $\mathcal{A}_{II}$ has access to them, i.e., $\mathcal{O}_{II} = \{\mathcal{O}_{pk}, \mathcal{O}_{sk}, \mathcal{O}_{Signcrypt}, \mathcal{O}_{Unsigncrypt}\}$, as follows:

Initialization: It is similar to the Initialization step of the proof of Lemma 2.

Queries: $\mathcal{B}$ responds all $\mathcal{A}_{II}$'s queries from $\mathcal{O}_{II}$ similar to them in the Phase 1 queries of the proof of Lemma 2.

Output: After a polynomially bounded number of queries (if $\mathcal{B}$ does not abort), $\mathcal{A}_{II}$ outputs a new valid signcryption $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, \sigma_5^*)$ on message M^* for ID_{S^*} with public key $pk_{ID_{S^*}} = (C^{\alpha x_{ID_{S^*}}}, A^{\alpha / x_{ID_{S^*}}})$ and ID_{R^*} with public key $pk_{ID_{R^*}} = (C^{\alpha x_{ID_{R^*}}}, A^{\alpha / x_{ID_{R^*}}})$. (Note that $(\sigma^*, ID_{S^*}, ID_{R^*})$ is not produced by the signcryption oracle.) Finally, when the Game IV between $\mathcal{A}_{II}$ and $\mathcal{B}$ terminates, $\mathcal{B}$ computes $m^* = H_1(\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*, ID_{R^*}, pk_{ID_{R^*}})$ and acts as follows:

- If $J_u(ID_{S^*}) \neq 0 \bmod q$ or $J_m(m^*) + \alpha x_{ID_{S^*}} h^* \neq 0 \bmod q$ (or the simulation fails in any steps), $\mathcal{B}$ aborts.
- If $J_u(ID_{S^*}) = 0 \bmod q$ and $J_m(m^*) + \alpha x_{ID_{S^*}} h^* = 0 \bmod q$ (and the simulation does not fail in any steps), $\mathcal{B}$ retrieves $x_{ID_{S^*}}$ and computes:

$$h^{a^3} = \left(\frac{\sigma_5^*}{(\sigma_4^*)^{K_u(ID_{S^*})}(\sigma_2^*)^{K_m(m^*)}}\right)^{\frac{1}{(\alpha x_{ID_{S^*}})^2}}.$$

Time and probability analyses are similar to those in the proof of lemma 3 except for $q_{psk} = 0$. ■

Theorem 2. The proposed scheme is existentially unforgeable under adaptive chosen message attack (according to Definition 2) in the standard model under the 3-CDHE assumption.

Proof. The proof is directly implied from Lemma 3 and Lemma 4. ■

Remark 3. In order to prove the security (i.e., confidentiality and unforgeability) of Liu *et al.*'s signcryption scheme against $\mathcal{A}_I$, they only considered the case that $\mathcal{A}_I$ can replace the public key of an entity $pk_{ID} = e(g_1, g_2)^{x_{ID}}$, with a correctly formed public key $pk'_{ID} = e(g_1, g_2)^{x'_{ID}}$. However, in the first proposed attack in [13, 14], $\mathcal{A}_I$ can cheat the sender and decrypt a ciphertext by replacing the public key of the receiver with $pk'_{ID_R} = e(g, g)^{x'_{ID_R}}$. Also, in the second proposed attack in [13], $\mathcal{A}_I$ can cheat the receiver and forge a valid signcryption by replacing the public key of the sender with $pk'_{ID_S} = e(g, g)^{x'_{ID_S}}$. Our scheme is robust against these attacks, since in the first step of both signcryption and unsigncryption phases, it is

Table 2. Comparisons Between the CLSC Schemes in the Standard Model

CLSC Scheme	Signcryption Operations	Unsigncryption Operations (for valid signcription)	Unsigncryption Operations(for invalid signcription)	Ciphertext Size	Secure Against Attacks in [12–14]
[5]	$1E_{G_2} + 3E_{G_1}$	$5P$	$3P$	$\|4G_1\| + \|1G_2\|$	No
[6]	$3E_{G_2} + 3E_{G_1}$	$5P + 2E_{G_2}$	$3P + 2E_{G_2}$	$\|4G_1\| + \|1G_2\|$	No
[7]	$3P + 3E_{G_2} + 6E_{G_1}$	$8P + 2E_{G_2} + 3E_{G_1}$	$6P + 2E_{G_2} + 1E_{G_1}$	$\|4G_1\| + \|1G_2\|$	Yes
[8]	$5P+1E_{G_2}+3E_{G_1}+1\phi$	$10P + 1\phi^{-1}$	$10P + 1\phi^{-1}$	$\|4G_1\| + \|1G_2\|$	Yes
[9]	$2P + 3E_{G_2} + 5E_{G_1} + 1\phi$	$7P + 2E_{G_1} + 1\phi^{-1}$	$7P + 2E_{G_1} + 1\phi^{-1}$	$\|4G_1\| + \|2G_2\|$	Yes
Ours	$2P + 7E_{G_1}$	$7P + 1E_{G_1}$	$5P + 1E_{G_1}$	$\|4G_1\| + \|1G_2\|$	Yes

checked whether the public keys are formed by the correct construction.

Remark 4. In two proposed attacks in [12], $\mathcal{A}_{II}$ generates the public parameters maliciously and wins Game II and Game IV. Our scheme is also robust against these attacks, because of the terms $(u' \prod_{i \in \mathcal{U}_{ID_S}} u_i)^{r_2}$ and $pk_{ID_{S,1}}^{hr_1}$ in the structure of σ_5. Note that even if $\mathcal{A}_{II}$ generates $u', v', u_i (i = 1, 2, \ldots, n_u)$ and $v_j (j = 1, 2, \ldots, n_m)$ maliciously such that she knows the discrete logarithms of them (like the attacks in [12]), she could not calculate $(u' \prod_{i \in \mathcal{U}_{ID_S}} u_i)^{r_2}$ and $pk_{ID_{S,1}}^{hr_1}$ and performs the attacks in [12].

4.3 Performance

To the best of our knowledge, only five CLSC schemes in the standard model have been proposed in the literature, till now [5–9]. We compare our scheme with these schemes in Table 2. E_{G_1}, E_{G_2} and P are the notations for an exponentiation in G_1, exponentiation in G_2 and pairing computations, respectively. $\phi : \Re \subseteq \{0,1\}^{n_m+n_u} \longrightarrow G_2$ is a bijection while ϕ^{-1} is its inverse [8, 9]. Also, $|aG|$ denotes the binary length of a elements in G. Since the schemes in [5, 6] are insecure, we will concentrate on the comparison of our scheme with the schemes in [7–9] with more details. Note that in the Unsigncryption phase of our scheme (as well as the schemes in [5–7]), the verification is applied before the encryption and as a result the unsigncryption operation decreases when the signcryption is invalid. But in the Unsigncryption phase of the schemes in [8, 9] the verification is applied after the encryption and as a result the unsigncryption operation is similar in both valid and invalid signcryptions. For an approximated comparison, consider most efficient up-to-date curves with and without pairing and count the cost of a modular exponentiation in the curve without pairing as 1. Denoting G_1 and G_2 the curves with the pairing, the cost of an exponentiation in G_1 is 3, the cost of an exponentiation in G_2 is 6, and the cost of a pairing is 8, approximately [18]. By these considerations, if the signcryption is valid, the total computation costs (for signcryption and unsigncryption operations) of the schemes in [7], [8] and [9] are approximately 145, 135 and 111, respectively (without considering ϕ and ϕ^{-1} computations) while the total computation cost of our scheme is approximately 96. Also, if the signcryption is invalid, the total computation costs of the schemes in [7], [8] and [9] are approximately 123, 135 and 111, respectively while the total computation cost of our scheme is approximately 80. So, our scheme is more efficient than other schemes in the sense of the computation cost especially for invalid signcryptions. As shown, the ciphertext size of our scheme is smaller than that of the scheme in [9]. In fact, the authors in [9] have tried to improve the computation cost of the scheme in [8], but their scheme is less efficient than the scheme in [8] in the sense of the ciphertext size. Our scheme is more efficient than the scheme in [8] (and even more efficient than the scheme in [9]) in the sense of the computation cost, while its ciphertext size is also kept unchanged in comparison with the scheme in [8].

As a final result, our scheme is not only robust against the attacks in [12–14], but also more efficient than all other secure CLSC schemes in the standard model in [7–9] both in terms of computation and communication costs.

5 Conclusion

An improved version of Liu *et al.*'s CLSC scheme was presented, which is robust against all proposed attacks to their scheme. The proposed scheme is semantically secure against adaptive chosen ciphertext attack under the $(\mathcal{S}_2, 5)$-BDHE-Set assumption and existentially unforgeable against adaptive chosen message attack under the 3-CDHE assumption in the standard model. Furthermore, the proposed scheme is more efficient than all other secure CLSC schemes in the standard

model proposed in the literature up to now.

References

[1] A. Shamir, Identity-Based Cryptosystem and Signature Scheme, In Advances in Cryptology, Crypto 84, Springer, LNCS, vol. 196, pp. 47-53, 1984.

[2] S.S. Al-Riyami and K. Paterson, Certificateless Public Key Cryptography, In Asiacrypt 2003, Springer, LNCS, vol. 2894, pp. 452-473, 2003.

[3] Y. Zheng, Digital Signcryption or How to Achieve Cost(Signature & Encryption) << Cost(Signature) + Cost(Encryption), In Advances in Cryptology, Crypto 97, Springer Berlin Heidelberg, pp. 165-179, 1997.

[4] M. Barbosa and P. Farshim, Certificateless signcryption, in Proceedings of the 2008 ACM Symposium on Information,Computer and Communications Security, pp. 369-372, 2008.

[5] Z. Liu,Y. Hu, X. Zhang and H. Ma, Certificateless signcryption scheme in the standard model, Information Siences, vol.180, no. 3, pp. 452-464, 2010.

[6] Z. Jin, Q. Wen, and H. Zhang, A supplement to Liu *et al.*'s certificateless signcryption scheme in the standard model, in IACR cryptology ePrint Archive, 2010.

[7] H. Xiong, Toward Certificateless Signcryption Scheme Without Random Oracles, in IACR Cryptology ePrint Archive, 2014.

[8] L. Cheng and Q. Wen, An Improved Certificateless Signcryption in the Standard Model, IJ Network Security, vol 17, no. 5, pp. 597-606, 2015.

[9] C. Zhou, G. Gao, and Z. Cui, Certificateless Signcryption in the Standard Model, Wireless Personal Communications, pp. 1-19, 2016.

[10] M. Luo, D. Huang and J. Hu, An Efficient Biometric Certificateless Signcryption Scheme, Journal of Computer, vol. 8, no. 7, pp. 1853-1860, 2013.

[11] J. Kar, A Novel Construction of Certificateless Signcryption Scheme for Smart Card, Case Studies in Secure Computing Achievements and Trends, CRC Press, Taylor and Francis, New York, pp. 437-456, 2014.

[12] J. Weng, G. X. Yao, R. H. Deng, M. R. Chen, and X. X. Li, Cryptanalysis of a certificateless signcryption scheme in the standard model, Information Sciences, vol. 181, no. 3, pp. 661-667, 2011.

[13] S. Miao, F. Zhang, S. S. Li, and Y. Mu, On security of a certificateless signcryption scheme, Information Sciences, vol. 232, pp. 475-481, 2013.

[14] S. S. D. Selvi, S. S. Vivek, and C. P. Rangan, Certificateless signcryption, in Cryptology ePrint Archive, Report 2010/92, 2010.

[15] F. Zhang, R. Safavi-Naini, and W. Susilo, An efficient signature scheme from bilinear pairings and its applications, In Public Key Cryptography PKC, Springer Berlin Heidelberg, pp. 27-290, Mar 2004.

[16] C. Gentry and S. Halevi, Hierarchical Identity Based Encryption with Polynomially Many Levels, In TCC 2009, Vol. 5444, pp. 437-456, Mar 2009.

[17] Y. Yuan and C. Wang, A Secure Certificateless Signature Scheme in the Standard Model, Journal of Computational Information Systems no.9, vol.11, pp. 4353-4362, 2013.

[18] F. Benhamouda, G. Couteau, D. Pointcheval, and H. Wee., Implicit zero-knowledge arguments and applications to the malicious setting, In Advances in Cryptology–CRYPTO 2015, Springer Berlin Heidelberg, pp. 107-129, 2015.

Parvin Rastegari received the B.S. and M.S. degrees in electrical engineering from department of electrical and computer engineering, Isfahan University of Technology in 2008 and 2011, respectively. She is currently a Ph.D. candidate in the same department. Her M.S. dissertation is in the field of information theory entitled "the redundancy of some source codes". Her research interests are digital signatures and advanced security protocols.

Mehdi Berenjkoub received the Ph.D. degree from department of electrical and computer engineering, Isfahan University of Technology in 2000. The title of his dissertation is two-party key distribution protocols in cryptography. He started his work in the same department as an assistant professor from that time. Graduate courses presented by him include fundamentals of cryptography, cryptographic protocols, network security, and intrusion detection. He has supervised more than a dozen M.S. students and Ph.D. candidates in related areas. He also was one of the founder members for Iranian Society of Cryptology in 2001. He has continued his cooperation with the society as an active member. He along with his colleagues established a research group on Security in Networks and Systems in IUT. He also is responsible for an established academic CSIRT in IUT. He is an associate professor and his current interested research topics are advanced security protocols, authentication protocols and network security.

The ISC Int'l Journal of Information Security
ISeCure

January 2017, Volume 9, Number 1 (p. 93)
http://www.isecure-journal.org

Persian Abstract

یک شمای امضارمز فاقد گواهی کارآمد در مدل استاندارد

پروین رستگاری[1] و مهدی برنج‌کوب[1]

[1] دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان، اصفهان، ایران

رمزنگاری کلید عمومی فاقد گواهی راه‌کاری مفید برای حل توأم مشکلات زیرساخت رمزنگاری کلید عمومی معمول (هزینهٔ بالای محاسبه، ذخیره‌سازی و مخابرهٔ گواهی‌ها) و مشکلات رمزنگاری کلید عمومی مبتنی بر شناسه (اطلاع مرکز تولید کلید از کلیدهای خصوصی تمامی موجودیت‌ها) به شمار می‌آید. یک شمای امضارمز عنصری مهم در پروتکل‌های امنیتی است که اهداف امضا و رمزنگاری را به‌صورت هم‌زمان برآورده می‌سازد. در سال ۲۰۱۰، Liu و همکاران طرحی را به‌عنوان اولین شمای امضارمز فاقد گواهی در مدل استاندارد پیشنهاد کردند، اما تاکنون حملات زیادی به شمای پیشنهادی آن‌ها ارائه شده است. در این مقاله، با بهبود طرح Liu و همکاران، یک شمای امضارمز فاقد گواهی در مدل استاندارد ارائه می‌شود که نه‌تنها در برابر حملات مذکور مقاوم است، بلکه از سایر شماهای امضارمز فاقد گواهی ارائه شده در مدل استاندارد کارآمدتر نیز می‌باشد.

واژه‌های کلیدی: رمزنگاری کلید عمومی فاقد گواهی، شمای امضارمز، مدل استاندارد، مدل اوراکل تصادفی.